



C++

Programming Step-by-Step

Asadullah Shah



IIUM PRESS

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA

C++ PROGRAMMING: STEP BY STEP

Editors

Asadullah Shah



IIUM Press

Published by:
IIUM Press
International Islamic University Malaysia

First Edition, 2011
©IIUM Press, IIUM

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without any prior written permission of the publisher.

Perpustakaan Negara Malaysia

Cataloguing-in-Publication Data

Bibliography p.
Includes Index
ISBN

ISBN: 978-967-418-090-4

Member of Majlis Penerbitan Ilmiah Malaysia -- MAPIM
(Malaysian Scholarly Publishing Council)

Printed by :
IIUM PRINTING SDN. BHD.
No. 1, Jalan Industri Batu Caves 1/3
Taman Perindustrian Batu Caves
Batu Caves Centre Point
68100 Batu Caves
Selangor Darul Ehsan

CONTENTS

DEDICATION	iii
PREFACE	viii
ACKNOWLEDGEMENT	ix
1. INTRODUCTION	
<i>Asadullah Shah and Assadullah Shaikh</i>	<i>1</i>
2. ARITHMETIC EXPRESSIONS AND DATA TYPES IN C++	
<i>Asadullah Shah and Assadullah Shaikh</i>	<i>5</i>
3. SENDING THE OUTPUT TO A PRINT FILE	
<i>Asadullah Shah and Assadullah Shaikh</i>	<i>11</i>
4. DECISION MAKING: IF-ELSE STATEMENTS AND RELATIONAL OPERATORS	
<i>Asadullah Shah and Assadullah Shaikh</i>	<i>17</i>
5. LOGICAL OPERATORS AND SWITCH STATEMENTS	
<i>Asadullah Shah and Assadullah Shaikh</i>	<i>25</i>
6. REVIEW, SUMMARY & BUILDING SKILL	
<i>Asadullah Shah and Khamran Khowaza</i>	<i>33</i>
7. ITERATIVE STRUCTURES	
<i>Asadullah Shah and Khamran Khowaza</i>	<i>39</i>

8. THE FOR LOOP

Asadullah Shah and Khamran Khowaza 49

9. THE DO-WHILE LOOP

Asadullah Shah and Khamran Khowaza 55

10. REVIEW OF VARIABLES, FORMATTING

Asadullah Shah and Khamran Khowaza 59

11. REVIEW OF ITERATIVE STRUCTURES

Asadullah Shah and Sumbul Khowaza 63

12. POST-TEST AND NESTED LOOPS

Asadullah Shah and Sumbul Khowaza 73

13. FUNCTIONS

Asadullah Shah and Sumbul Khowaza 83

14. CALL-BY-VALUE AND REFERENCE

Asadullah Shah and Sumbul Khowaza 91

15. MORE ON FUNCTIONS

Asadullah Shah and Sumbul Khowaza 99

16. STRUCTURES (STRUCT) AND FILES

Asadullah Shah and Muniba Shaikh 111

17. ARRAYS

Asadullah Shah and Muniba Shaikh 119

18. EXERCISE OF ARRAY

Asadullah Shah and Muniba Shaikh 127

19. READ DATA FROM A FILE	
<i>Asadullah Shah and Mumiba Shaikh</i>	137
20. OBJECT ORIENTED PROGRAMMING	
<i>Asadullah Shah and Mumiba Shaikh</i>	143
21. SELECTION SORTING	
<i>Asadullah Shah and Syed Ifthar Ali</i>	153
22. BUBBLE SORT ALGORITHM	
<i>Asadullah Shah and Syed Ifthar Ali</i>	161
23. REVIEW OF ARRAYS	
<i>Asadullah Shah and Syed Ifthar Ali</i>	167
24. LINEAR SEARCHING	
<i>Asadullah Shah and Syed Ifthar Ali</i>	179
25. BINARY SEARCH	
<i>Asadullah Shah and Syed Ifthar Ali</i>	189
26. VECTOR CLASS	
<i>Asadullah Shah and Ejaz Ahmed</i>	199
27. POINTERS	
<i>Asadullah Shah and Ejaz Ahmed</i>	203
28. FUNCTION POINTERS	
<i>Asadullah Shah and Ejaz Ahmed</i>	213
29. POLYMORPHISM AND VIRTUAL FUNCTIONS	
<i>Asadullah Shah and Ejaz Ahmed</i>	219

30. C++ REFERENCES	
<i>Asadullah Shah and Ejaz Ahmed</i>	223
31. CONST CORRECTNESS	
<i>Asadullah Shah and Osama Mahfooz</i>	229
32. MORE ON CONST KEYWORDS	
<i>Asadullah Shah and Osama Mahfooz</i>	235
33. GOTO STATEMENT	
<i>Asadullah Shah and Osama Mahfooz</i>	241
34. HANDLING ERRORS IN C++	
<i>Asadullah Shah and Osama Mahfooz</i>	249
35. STATIC: THE MULTIPURPOSE KEYWORD	
<i>Asadullah Shah and Osama Mahfooz</i>	253

18. EXERCISE OF ARRAY

Asadullah Shah and Muniba Shaikh

Department of Computer Science, Faculty of Information and
Communication Technology, International Islamic University Malaysia,
Malaysia

Abstract

Arrays in C++ are treated as static lists. Arrays are used to store and spare storage space for identical data items in memory areas. The size of an array must be equal to the number of items to be deposited in that storage area. Smaller size arrays create data overflow problems and data items may corrupt the other users data out the reserved area. Practicing array in programming is of utmost importance and value. This chapter is reserved for the array program practices.

18.1 Introduction

A simple programs given in Figure 18.1 is to calculate variance among data items. In order to calculate it steps are as follows:

1. Find the mean (average) of all data item.
2. Find the difference of each data item from the mean.
3. Square the differences (item #2).
4. Find the sum of all squared differences (item #3).
5. Find the variance by dividing the sum (item #4) by the number of scores minus one.